

1 **KEY AGREEMENT AND TRANSPORT PROTOCOL**

2 This application is a continuation-in-part of United States Application
3 No. 98/426,090.

4 The present invention relates to key agreement protocols for transfer and
5 authentication of encryption keys.

6 To retain privacy during the exchange of information it is well known
7 to encrypt data using a key. The key must be chosen so that the correspondents are
8 able to encrypt and decrypt messages but such that an interceptor cannot determine the
9 contents of the message.

10 In a secret key cryptographic protocol, the correspondents share a
11 common key that is secret to them. This requires the key to be agreed upon between
12 the correspondents and for provision to be made to maintain the secrecy of the key
13 and provide for change of the key should the underlying security be compromised.

14 Public key cryptographic protocols were first proposed in 1976 by
15 Diffie-Hellman and utilized a public key made available to all potential
16 correspondents and a private key known only to the intended recipient. The public
17 and private keys are related such that a message encrypted with the public key of a
18 recipient can be readily decrypted with the private key but the private key cannot be
19 derived from the knowledge of the plaintext, ciphertext and public key.

20 Key establishment is the process by which two (or more) parties
21 establish a shared secret key, called the session key. The session key is subsequently
22 used to achieve some cryptographic goal, such as privacy. There are two kinds of key
23 agreement protocol; key transport protocols in which a key is created by one party and
24 securely transmitted to the second party; and key agreement protocols, in which both
25 parties contribute information which jointly establish the shared secret key. The
26 number of message exchanges required between the parties is called the number of
27 passes. A key establishment protocol is said to provide implicit key authentication (or
28 simply key authentication) if one party is assured that no other party aside from a
29 specially identified second party may learn the value of the session key. The property
30 of implicit key authentication does not necessarily mean that the second party actually

1 possesses the session key. A key establishment protocol is said to provide key
2 confirmation if one party is assured that a specially identified second party actually
3 has possession of a particular session key. If the authentication is provided to both
4 parties involved in the protocol, then the key authentication is said to be mutual if
5 provided to only one party, the authentication is said to be unilateral.

6 There are various prior proposals which claim to provide implicit key
7 authentication.

8 Examples include the Nyberg-Rueppel one-pass protocol and the
9 Matsumoto-Takashima-Imai (MTI) and the Goss and Yacobi two-pass protocols for
10 key agreement.

11 The prior proposals ensure that transmissions between correspondents
12 to establish a common key are secure and that an interloper cannot retrieve the session
13 key and decrypt the ciphertext. In this way security for sensitive transactions such as
14 transfer of funds is provided.

15 For example, the MTI/A0 key agreement protocol establishes a shared
16 secret K, known to the two correspondents, in the following manner:-

17 1. During initial, one-time setup, key generation and publication is
18 undertaken by selecting and publishing an appropriate system prime p and generator
19 in a manner guaranteeing authenticity. Correspondent A selects as a long-term private
20 key a random integer " a ", $1 \leq a \leq p-2$, and computes a long-term public key $z_A = \alpha^a \bmod$
21 p . B generates analogous keys b, z_B . A and B have access to authenticated copies of
22 each other's long-term public key.

23 2. The protocol requires the exchange of the following messages.

24 $A \rightarrow B: \alpha^x \bmod p$ (1)

25 $A \leftarrow B: \alpha^y \bmod p$ (2)

26 The values of x and y remain secure during such transmissions as it is
27 impractical to determine the exponent even when the value of α and the
28 exponentiation is known provided of course that p is chosen sufficiently large.

29 3. To implement the protocol the following steps are performed each time
30 a shared key is required.

- 1 (a) A chooses a random integer $x, 1 \leq x \leq p-2$, and sends B message
2 (1) i.e. $\alpha^x \bmod p$.
3 (b) B chooses a random integer $y, 1 \leq y \leq p-2$, and sends A message
4 (2) i.e. $\alpha^y \bmod p$.
5 (c) A computes the key $K = (\alpha^y)^a z_B^x \bmod p$.
6 (d) B computes the key $K = (\alpha^x)^b z_A^y \bmod p$.
7 (e) Both share the key $K = \alpha^{bx+ay}$.

8
9 In order to compute the key K , A must use his secret key a and the
10 random integer x , both of which are known only to him. Similarly B must use her
11 secret key b and random integer y to compute the session key K . Provided the secret
12 keys a, b remain uncompromised, an interloper cannot generate a session key identical
13 to the other correspondent. Accordingly, any ciphertext will not be decipherable by
14 both correspondents.

15 As such this and related protocols have been considered satisfactory for
16 key establishment and resistant to conventional eavesdropping or man-in-the-middle
17 attacks.

18 In some circumstances it may be advantageous for an adversary to
19 mislead one correspondent as to the true identity of the other correspondent.

20 In such an attack an active adversary or interloper E modifies messages
21 exchanged between A and B, with the result that B believes that he shares a key K
22 with E while A believes that she shares the same key K with B. Even though E does
23 not learn the value of K the misinformation as to the identity of the correspondents
24 may be useful.

25 A practical scenario where such an attack may be launched
26 successfully is the following. Suppose that B is a bank branch and A is an account
27 holder. Certificates are issued by the bank headquarters and within the certificate is
28 the account information of the holder. Suppose that the protocol for electronic deposit
29 of funds is to exchange a key with a bank branch via a mutually authenticated key
30 agreement. Once B has authenticated the transmitting entity, encrypted funds are

1 deposited to the account number in the certificate. If no further authentication is done
2 in the encrypted deposit message (which might be the case to save bandwidth) then
3 the deposit will be made to E's account.

4 It is therefore an object of the present invention to provide a protocol in
5 which the above disadvantages are obviated or mitigated.

6 According therefore to the present invention there is provided a method
7 of authenticating a pair of correspondents A,B to permit exchange of information
8 therebetween, each of said correspondents having a respective private key a,b and a
9 public key p_A,p_B derived from a generator α and respective ones of said private keys
10 a,b , said method including the steps of

11 i) a first of said correspondents A selecting a first random integer x and
12 exponentiating a function $f(\alpha)$ including said generator to a power $g^{(x)}$ to provide a
13 first exponentiated function $f(\alpha)^{g^{(x)}}$;

14 ii) said first correspondent A forwarding to a second correspondent B a message
15 including said first exponentiated function $f(\alpha)^{g^{(x)}}$;

16 iii) said correspondent B selecting a second random integer y and exponentiating a
17 function $f(\alpha)$ including said generator to a power $g^{(y)}$ to provide a second
18 exponentiated function $f(\alpha)^{g^{(y)}}$;

19 iv) said second correspondent B constructing a session key K from information
20 made public by said first correspondent A and information that is private to said
21 second correspondent B, said session key also being constructible by said first
22 correspondent A for information made public by B and information that is private to
23 said first correspondent A;

24 v) said second correspondent B generating a value h of a function $F[\delta,K]$
25 where $F[\delta,K]$ denotes a cryptographic function applied conjointly to δ and K and
26 where δ is a subset of the public information provided by B thereby to bind the values
27 of δ and K ;

28 vi) said second of said correspondents B forwarding a message to said first
29 correspondent A including said second exponential function $f(\alpha)^{g^{(y)}}$ and said value h
30 of said cryptographic function $F[\delta,K]$;

1 vii) said first correspondent receiving said message and computing a session key
2 K' from information made public by said second correspondent B and private to said
3 first correspondent A;

4 viii) said first correspondent A computing a value h' of a cryptographic function
5 h,h' F[δ ,K']; and

6 ix) comparing said values obtained from said cryptographic functions F to
7 confirm their correspondence.

8 As the session key K can only be generated using information that is
9 private to either A or B, the binding of K with δ with the cryptographic function h
10 prevents E from extracting K or interjecting a new value function that will correspond
11 to that obtained by A.

12 Embodiments of the invention will now be described by way of
13 example only with reference to the accompanying drawings in which.

14 Figure 1 is a schematic representation of a data communication system.

15 Figures 2 through 7 are schematic representations of implementations
16 of different protocols.

17 Referring therefore to Figure 1, a pair of correspondents, 10,12,
18 denoted as correspondent A and correspondent B, exchange information over a
19 communication channel 14. A cryptographic unit 16,18 is interposed between each of
20 the correspondents 10,12 and the channel 14. A key 20 is associated with each of the
21 cryptographic units 16,18 to convert plaintext carried between each unit 16,18 and its
22 respective correspondent 10,12 into ciphertext carried on the channel 14.

23 In operation, a message generated by correspondent A, 10, is encrypted
24 by the unit 16 with the key 20 and transmitted as ciphertext over channel 14 to the
25 unit 18.

26 The key 20 operates upon the ciphertext in the unit 18 to generate a
27 plaintext message for the correspondent B, 12. Provided the keys 20 correspond, the
28 message received by the correspondent 12 will be that sent by the correspondent 10.

29 In order for the system shown in Figure 1 to operate it is necessary for
30 the keys 20 to be identical and therefore a key agreement protocol is established that

1 allows the transfer of information in a public manner to establish the identical keys. A
2 number of protocols are available for such key generation and embodiments of the
3 present invention will be described below in the context of modifications of existing
4 protocols.

5 A commonly used set of protocols are collectively known as the
6 Matsumoto-Takahima-Imai or "MTI" key agreement protocols, and are variants of
7 the Diffie-Hellman key exchange. Their purpose is for parties A and B to establish a
8 secret session key K.

9 The system parameters for these protocols are a prime number p and a
10 generator α of the multiplicative group

11 . Correspondent A has private key a and public key $p_A = \alpha^a$. Correspondent B has
12 private key b and public key $p_B = \alpha^b$. In all four protocols exemplified below, $text_A$
13 refers to a string of information that identifies party A. If the other correspondent B
14 possesses an authentic copy of correspondent A's public key, then $text_A$ will contain
15 A's public-key certificate, issued by a trusted center; correspondent B can use his
16 authentic copy of the trusted center's public key to verify correspondent A's certificate,
17 hence obtaining an authentic copy of correspondent A's public key.

18 In each example below it is assumed that an interloper E wishes to
19 have messages from A identified as having originated from E herself. To accomplish
20 this, E selects a random integer e, $1 \leq e \leq p-2$, computes $p_E = (p_A)^e = \alpha^{ae} \bmod p$, and gets
21 this certified as her public key. E does not know the exponent ae, although she knows
22 e. By substituting $text_E$ for $text_A$, the correspondent B will assume that the message
23 originates from E rather than A and use E's public key to generate the session key K.
24 E also intercepts the message from B and uses his secret random integer e to modify
25 its contents. A will then use that information to generate the same session key
26 allowing A to communicate with B.

27 The present invention is exemplified by modifications to 4 of the
28 family of MTI protocols which foil this new attack thereby achieving the desired
29 property of mutual implicit authentication. In the modified protocols exemplified
30 below $F(X,Y)$ denotes a cryptographic function applied to a string derived from x and

- 1 y. Typically and as exemplified a hash function, such as the NIST "Secure Hash
2 Algorithm"(SHA-1), is applied to the string obtained by concatenating X and Y but it
3 will be understood that other cryptographic functions may be used.

4 **Example 1 - MTI/A0 protocol**

5 The existing protocol operates as follows:-

- 6 1. Correspondent A generates a random integer
7 x , $1 \leq x \leq p-2$, computes α^x , and sends $\{\alpha^x, \text{text}_A\}$ to party B.
- 8 2. Correspondent B generates a random integer
9 y , $1 \leq y \leq p-2$, computes α^y , and sends $\{\alpha^y, \text{text}_B\}$ to party A.
- 10 3. Correspondent A computes $K = (\alpha^y)^a (p_B)^x = \alpha^{ay+bx}$.
- 11 4. Correspondent B computes $K = (\alpha^x)^b (p_A)^y = \alpha^{ay+bx}$.

12

13 A common key K is thus obtained. However, with this arrangement,
14 interloper E may have messages generated by correspondent A identified as having
15 originated from E in the following manner.

- 16 1. E intercepts A's message $\{\alpha^x, \text{text}_A\}$ and replaces it with $\{\alpha^x, \text{text}_E\}$.
17 The provision of the message text_E identifies the message as having originated at E.
- 18 2. B sends $\{\alpha^y, \text{text}_B\}$ to E, who then forwards $\{(\alpha^y)^e, \text{text}_B\}$ to A. Since A
19 receives text_B , he assumes the message originates at B and, as he does not know the
20 value of y , assumes that α^{ye} is valid information.
- 21 3. A computes $K = (\alpha^{ey})^a (p_B)^x = \alpha^{aey+bx}$.
- 22 4. B computes $K = (\alpha^x)^b (p_E)^y = \alpha^{aey+bx}$.
- 23 5. A and B now share the key K, even though B believes he shares a key
24 with E.

25

26 Accordingly any further transactions from A to B will be considered by
27 B to have originated at E. B will act accordingly crediting instruction to E. Even
28 though the interloper E does not learn the value of the session key K nevertheless the
29 assumption that the message originates at E may be valuable and achieve the desired
30 effect.

- 1 To avoid this problem, the protocol is modified as follows:-
- 2 1. A generates a random integer $x, 1 \leq x \leq p-2$, computes α^x , and sends
- 3 $\{\alpha^x, \text{text}_A\}$ to party B.
- 4 2. B generates a random integer $y, 1 \leq y \leq p-2$, and computes α^y , K
- 5 $= (\alpha^x)^y = \alpha^{xy}$, and a value h of cryptographic hash function $F(\alpha^y, \alpha^{xy})$ which is a
- 6 function of public information δ and the key K . B sends $\{\alpha^y, h, \text{text}_B\}$ to party A.
- 7 3. A computes $K = (\alpha^y)^x = \alpha^{xy}$. A also computes a value h' of
- 8 cryptographic hash function $F(\alpha^y, K)$ and verifies that this value is equal to h .

9

10 If E attempts to interpose her identification, text_E , the attack fails on

11 the modified protocols because in each case B sends the hash value $F(\delta, K)$, where δ is

12 B's random exponential, α^y , thereby binding together the values of δ and K . E cannot

13 subsequently replace the value of δ with δ^e and compute $F(\delta^e, K)$ since E does not

14 know K . Even though E knows α^y , this is not sufficient to extract K from the hash

15 value h . Accordingly, even if E interposes the value α^{ye} so that the keys will agree,

16 the values h, h' will not.

17

18 Example 2 - MTI/B0 protocol

19 In this protocol,

- 20 1. A generates a random integer $x, 1 \leq x \leq p-2$, computes $(p_B)^x = \alpha^{bx}$, and
- 21 sends $\{\alpha^{bx}, \text{text}_A\}$ to party B.
- 22 2. B generates a random integer $y, 1 \leq y \leq p-2$, computes $(p_A)^y = \alpha^{ay}$, and
- 23 sends $\{\alpha^{ay}, \text{text}_B\}$ to party A.
- 24 3. A computes $K = (\alpha^{ay})^x = \alpha^{xy}$
- 25 4. B computes $K = (\alpha^{bx})^y = \alpha^{xy}$

26

27 This protocol is vulnerable to the interloper E if,

- 28 1. E replaces A's message $\{\alpha^{bx}, \text{text}_A\}$ with $\{\alpha^{bx}, \text{text}_E\}$ to identify herself
- 29 as the originator to the message.
- 30 2. B sends $\{(p_E)^y, \text{text}_B\}$ to E, who then computes

- 1 $((P_E)^y)^{a'} = \alpha^{ay}$ and forwards $\{\alpha^{ay}, \text{text}_B\}$ to A.
- 2 3. A computes $K = (\alpha^{ay})^{a'} \alpha^x = \alpha^{x+ay}$
- 3 4. B computes $K = (\alpha^{bx})^{b'} \alpha^y = \alpha^{x+ay}$
- 4 5. A and B now share the key K, even though B believes he shares a key
- 5 with E.

6

7 This protocol may be modified to resist E's attack as follows.

- 8 1. A generates a random integer $x, 1 \leq x \leq p-2$, computes $(p_B)^x = \alpha^{bx}$, and
- 9 sends $\{\alpha^{bx}, \text{text}_A\}$ to party B.
- 10 2. B generates a random integer $y, 1 \leq y \leq p-2$, and computes $(p_A)^y = \alpha^{ay}$,
- 11 $K = (\alpha^{bx})^y = \alpha^{x+by}$, and the value h of hash function $F(\alpha^y = \alpha^{x+by})$. B
- 12 sends $\{\alpha^{ay}, h, \text{text}_B\}$ to A.
- 13 3. A computes $K = (\alpha^{ay})^x = \alpha^{x+ay}$. A also computes the value h' of hash
- 14 function $F(\alpha^{ay}, K)$ and verifies that this value is equal to h .

15 Once again, E cannot determine the session key K and so cannot

16 generate a new value of the hash function to maintain the deception.

17 Example 3 - MTI/CO protocol

18 This protocol operates as follows:-

- 19 1. A generates a random integer $x, 1 \leq x \leq p-2$, computes $(p_B)^x = \alpha^{bx}$, and
- 20 sends $\{\alpha^{bx}, \text{text}_A\}$ to party B.
- 21 2. B generates a random integer $y, 1 \leq y \leq p-2$, computes $(p_A)^y = \alpha^{ay}$, and
- 22 sends $\{\alpha^{ay}, \text{text}_B\}$ to party A.
- 23 3. A computes $K = (\alpha^{ay})^{a'x} = \alpha^{xy}$
- 24 4. B computes $K = (\alpha^{bx})^{b'y} = \alpha^{xy}$

25

26 The interloper E may interpose her identity as follows:-

- 27 1. E replaces A's message $\{\alpha^{bx}, \text{text}_A\}$ with $\{\alpha^{bx}, \text{text}_E\}$.
- 28 2. B sends $\{(p_E)^y, \text{text}_B\}$ to E, who then computes $((p_E)^y)^{e-1} = \alpha^{ay}$ and
- 29 forwards $\{\alpha^{ay}, \text{text}_B\}$ to A.

- 1 3. A computes $K = (\alpha^{ay})^{a^{-1}x} = \alpha^{xy}$
 2 4. B computes $K = (\alpha^{bx})^{b^{-1}y} = \alpha^{xy}$
 3 5. A and B now share the key K, even though B believes he shares a key
 4 with E.

5
 6 To avoid this attack protocol is modified as follows:-

- 7 1. A generates a random integer $x, 1 \leq x \leq p-2$, computes $(p_B)^x = \alpha^{bx}$, and
 8 sends $\{\alpha^{bx}, \text{text}_A\}$ to party B.
 9 2. B generates a random integer $y, 1 \leq y \leq p-2$, and computes
 10 $(p_A)^y = \alpha^{ay}$, $K = (\alpha^{bx})^{b^{-1}y} = \alpha^{xy}$, and value
 11 h of hash function $F(\alpha^{ay}, \alpha^{xy})$. B sends $\{\alpha^{ay}, h, \text{text}_B\}$ to party A.
 12 3. A computes $K = (\alpha^{ay})^{a^{-1}x} = \alpha^{xy}$. A also computes the value h' of
 13 $F(\alpha^{ay}, K)$ and verifies that this value is equal to h.

1 Example 4 - MTI/C1 protocol**2 In this protocol:-**

- 3 1. A generates a random integer $x, 1 \leq x \leq p-2$, computes $(p_B)^{ax} = \alpha^{abx}$, and
- 4 sends $\{\alpha^{abx}, \text{text}_A\}$ to party B.
- 5 2. B generates a random integer $y, 1 \leq y \leq p-2$, computes $(p_A)^{by} = \alpha^{aby}$, and
- 6 sends $\{\alpha^{aby}, \text{text}_B\}$ to party A.
- 7 3. A computes $K = (\alpha^{aby})^x = \alpha^{abxy}$.
- 8 4. B computes $K = (\alpha^{abx})^y = \alpha^{abxy}$.

9 E can act as an interloper as follows:-

- 11 1. E replaces A's message $\{\alpha^{abx}, \text{text}_A\}$ with $\{\alpha^{abx}, \text{text}_E\}$.
- 12 2. B sends $\{(p_E)^{by}, \text{text}_B\}$ to E, who then computes $((p_E)^{by})^{c-1} = \alpha^{uby}$ and
- 13 forwards $\{\alpha^{uby}, \text{text}_B\}$ to A.
- 14 3. A computes $K = (\alpha^{uby})^x = \alpha^{ubxy}$.
- 15 4. B computes $K = (\alpha^{abx})^y = \alpha^{abxy}$.
- 16 5. A and B now share the key K, even though B believes he shares a key
- 17 with E.

18 To avoid this, the protocol is modified as follows:-

- 20 1. A generates a random integer $x, 1 \leq x \leq p-2$, computes $(p_B)^{ax} = \alpha^{abx}$, and
- 21 sends $\{\alpha^{abx}, \text{text}_A\}$ to party B.
- 22 2. B generates a random integer $y, 1 \leq y \leq p-2$, and computes $(p_A)^{by} = \alpha^{aby}$, K
- 23 $= (\alpha^{abx})^y = \alpha^{abxy}$, and
- 24 $h = F(\alpha^{aby}, \alpha^{abxy})$. B sends $\{\alpha^{aby}, h, \text{text}_B\}$ to party A.
- 25 3. A computes $K = (\alpha^{aby})^x = \alpha^{abxy}$. A also computes
- 26 $h' = F(\alpha^{aby}, K)$ and verifies that this value is equal to h.

27

28 In each of the modified protocols discussed above, key confirmation

29 from B to A is provided.

30 As noted above instead of F being a cryptographic hash function other

1 functions could be used. For example, an option available is to choose
2 $F = E_K$, where E is the encryption function of a suitable symmetric-key encryption
3 scheme, and K is the session key established. Because E cannot generate the session
4 key K , it is similarly not able to generate the value of the function F and therefore
5 cannot interpose for the correspondent A .

6 The technique described above can be applied to other similar key
7 exchange protocols, including all of the 3 infinite classes of MTI protocols called
8 MTI-A(k), MTI-B(k) and MTI-C(k).

9 The Goss authenticated key exchange protocol is similar to the
10 MTI/A0 protocol, except that the session key is the bitwise exclusive-OR of α^{ay} and
11 α^{bx} ; that is $K = \alpha^{ay} \oplus \alpha^{bx}$ instead of being the product of α^{ay} and α^{bx} . Hence the attack
12 on the MTI/A0 protocol and its modification can be extended in a straightforward
13 manner to the case of the Goss protocol.

14 Similarly Yacobi's authenticated key exchange protocol is exactly the
15 same as the MTI/A0 protocol, except that α is an element of the group of units
16 \mathbb{Z}_n^* , where n is the product of 2 large primes. Again, the attack on the MTI/A0
17 protocol and its modification can be extended in a straightforward manner to the case
18 of the Goss protocol.

19 A further way of foiling the interposition of E is to require that each
20 entity prove to a trusted center that it knows the exponent of α that produces its public
21 key P , before the center issues a certificate for the public key. Because E only knows
22 "e" and not "ae" it would not meet this requirement. This can be achieved through
23 zero knowledge techniques to protect the secrecy of the private keys but also requires
24 the availability of a trusted centre which may not be convenient.

25 Each of the above examples has been described with a 2 pass protocol
26 for key authentication. One pass protocols also exist to establish a key between
27 correspondents and may be similarly vulnerable.

28 As an example the Nyberg-Rueppel one pass key agreement protocol
29 will be described and a modification proposed.

30 The purpose of this protocol is for party A and party B to agree upon a

1 secret session key K.

2 The system parameters for these protocols are a prime number p and a
3 generator α of the multiplicative group $\alpha \in Z_p^*$. User A has private key a and public
4 key $p_A = \alpha^a$. User B has private key b and public key $p_B = \alpha^b$.

5 1. A selects random integers x and t, $1 \leq x, t \leq p-2$.

6 2. B recovers the value $\alpha^x \bmod p$ by computing $\alpha^x (p_A)^t \bmod p$ and then
7 computes the shared session key $K = (r \alpha^x)^{b^{-1}} = \alpha^t \bmod p$.

8

9 If interloper E wishes to have messages from A identified as having
10 originated from herself, E selects a random integer e, $1 \leq e \leq p-2$, computes $p_E = \alpha^e$, and
11 gets this certified as her public key.

12 1. E intercepts A's message $\{r, s, \text{text}_A\}$ and computes $\alpha^x = \alpha^s (p_A)^r$ and α^{bt}
13 $= r \alpha^x$.

14 2. E then selects a random integer x' , $1 \leq x' \leq p-2$, computes $r' = \alpha^{bt} \alpha^{-x'}$
15 $\bmod p$ and $s' = x' - r'e \bmod (p-1)$.

16 3. E sends $\{r', s', \text{text}_E\}$ to B.

17 4. B recovers the value α^{bt} by computing $\alpha^{bt} (p_E)^{r'} \bmod p$ and then
18 computes $K' = (r' \alpha^{bt})^{b^{-1}} = \alpha^t \bmod p$.

19 5. A and B now share the key K, even though B believes he shares a key
20 with E.

21

22 To foil such an attack the protocol is modified by requiring A to also
23 transmit a value h of $F(p_A, K)$, where F is a hash function, an encryption function of a
24 symmetric-key system with key K or other suitable cryptographic function. The
25 modified protocol is the following.

26 1. A selects random integers x and t, $1 \leq x, t \leq p-2$.

27 2. A computes $r = (p_B)^t \alpha^{-x} \bmod p$, $s = x - ra \bmod$

28 $(p-1)$, session key $K = \alpha^t \bmod p$ and the value h of hash function

29 $F(p_A, K)$. A sends $\{r, s, h, \text{text}_A\}$ to B.

1 3. B recovers the value $\alpha^x \bmod p$ by computing $\alpha^s(p_A)^r \bmod p$ and then
 2 computes the shared session key $K=(r\alpha^x)^{b^{-1}} = \alpha^1 \bmod p$. B also
 3 computes the value h' of function $F(p_A, K)$ and verifies that this value is
 4 equal to h .

5 Again therefore by binding together the public information π and the
 6 session key K in the hash function, the interposition of E will not result in identical
 7 hash functions h, h' .

8 In each case it can be seen that a relatively simple modification to the
 9 protocols involving the binding of public and private information in a cryptographic
 10 function foils the interposition of interloper E .

11 All the protocols discussed above have been described in the setting of
 12 the multiplicative group $\alpha \in Z_p^*$. However, they can all be easily modified to work in
 13 any finite group in which the discrete logarithm problem appears intractable. Suitable
 14 choices include the multiplicative group of a finite field (in particular the finite field
 15 $GF(2^n)$), subgroups of $\alpha \in Z_p^*$ of order q , and the group of points on an elliptic curve
 16 defined over a finite field. In each case an appropriate generator α will be used to
 17 define the public keys.

18 The protocols discussed above can also be modified in a
 19 straightforward way to handle the situation where each user picks their own system
 20 parameters p and α (or analogous parameters if a group other than Z_p^* is used).

21 Further implementations are shown schematically in figures 2 through 7. A
 22 different notation is utilized but it will be understood that this notation may be
 23 mapped to that of the previous embodiments.

24

25 Referring to figure 2, a mutual public key authenticated key agreement protocol is
 26 complemented between a correspondent A shown on the left hand side of the figure
 27 and a correspondent B shown on the right hand side. Correspondent A has a public-
 28 private key pair P_A, S_A respectively and similarly correspondent B has a public private

1 Key pair P_B, S_B .

2

3 As a first step, correspondent A generates a session private key as a random number
4 RND_A and computes a corresponding public session key $G_A = F_A(RND_A)$. The
5 function F_A is a cryptographic one way function, typically an exponentiation by the
6 group generator, such as a point multiplication in an elliptic curve cryptosystem.

7

8 The public session key G_A is forwarded to correspondent B who generates
9 corresponding parameters of a session private key RND_B and the exponent G_B .

10

11 The correspondent B computes a session key K as a function of A's public
12 information G_A, P_A AND B's private information RND_B, S_B . A corresponding key K'
13 can be computed by A using the private information of A and the public information
14 of B namely $f(RND_A, G_B, S_A, P_B)$.

15

16 After correspondent B has generated the key K , he compiles a string $(G_A // G_B // Id_A)$
17 where Id_A is a string that identifies A. The concatenated string is hashed with a
18 cryptographic function h_K which is a keyed hash function that uses the key K to yield a
19 string $hash_B$.

20

21 The string $hash_B$ is forwarded to correspondent A together with Id_A and G_B .

22

23 Upon receipt of the message from B, correspondent A computes the key K' as
24 described above. Correspondent A also computes a hash, $hashverify_B$ from the string
25 $(G_B // G_A // Id_A)$ using the hash function keyed by the key K' . correspondent A checks
26 that the hashes verify to confirm the identity of the keys K, K' .

27

28 Correspondent A then computes a hash h_K using the key K on the string $(G_A // G_B // Id_B)$
29 and forwards that together with Id_B correspondent B. Correspondent B similarly
30 computes a $hashverify_A$ using the keyed hash function h_K on the same string and

1 verifies that $hash_A = hashverify_A$.

2

3 A similar protocol is shown in figure 3 to implement a mutual symmetric key
4 authentication protocol. In this protocol the correspondents share a key K obtained
5 over a secure channel. The correspondents A,B, each generate a random integer which
6 is used as the session public key of A and B respectively. Thereafter the exchange of
7 information and verification proceeds as above with respect to figure 2 with the
8 shared secret key being utilised in the keyed hash functions.

9

10 A full mutual public key authenticated protocol is shown in figure 4. An initial
11 exchange of the public keys P_A, P_B is performed over an authenticated channel
12 followed by the exchange of information as shown in the protocol of figure 4. In this
13 case the correspondent A sends G_A computed as described above with respect to
14 figure 2, together with a string x_2 that A wants confirmation of receipt by B.
15 Correspondent B computes the key K as in figure 2 and also generates a pair of strings
16 y_1, y_2 which B wants to have authenticated by A and receipt confirmed by A
17 respectively. The strings are sent to A with the hash $hash_B$ and identity Id_A . The hash
18 $hash_B$ is performed on a string including the message x_2 and the string y_1 wants
19 authenticated.

20

21 Correspondent A computes the key K and verifies the hash as before. This also
22 confirms receipt of x_2 by B.

23

24 Correspondent A in turn generates strings z_1, z_2 where z_1 is a string that A wants
25 authenticated by B and z_2 is a string that may be used in a subsequent stage of the
26 protocol described below. The strings, z_1 and y_2 together with the identifying
27 information of B, Id_B , are included in the string that is hashed with the key K to
28 provide the string $hash_A$. this is sent together with the identity of B and the strings
29 z_1, z_2 to the correspondent B who can verify the hashes as before, thereby confirming
30 receipt of y_2 and authenticating z_1 .

1

2 Thus the exchange of information is exchanged in an authenticated manner and a
3 common key obtained that allows subsequent exchange of correspondence on a secure
4 channel.

5

6 With the protocol described in figure 4 it is possible to implement a mutual public key
7 authenticated key agreement protocol by letting the strings x_2, y_1, y_2, z_1, z_2 all be empty
8 strings. Alternatively, a mutual public key authenticated key agreement protocol with
9 key transport can be implemented by using x_2 as a string that is assumed to represent
10 $E_K(k)$. Correspondent B can compute the value of K and hence retrieve the notional
11 value of k from the string. He can use this as his CRP,. The values of y_1 may be used
12 to represent $E_K(k_{21})$ and z_1 as $E_K(k_{12})$ where k_{21} and k_{12} are different keys for
13 communication or other secret information to be shared between the correspondents.
14 In this case y_1 and z_2 are empty strings. In this way there is a key agreement on a
15 shared key K_{AB} together with authenticated key transport of the keys k_{21} and
16 k_{12} between the correspondents. Moreover, if additional information is provided in the
17 x_2 and y_2 then confirmation of proper receipt is also obtained.

18

19 The protocol of figure 4 may also be used to increase efficiency in successive sessions
20 by using the string z_2 to pass the information exchanged in the first pass of the next
21 session. Thus as shown in figure 5, the string G_A, x_2 is sent as z_2 in the previous
22 session. The protocol then proceeds from correspondent B as before. Correspondent B
23 may also take advantage of this facility by including the information G_B, y_1 for the next
24 session in the exchange as y_2 .

25

26 The mutual public key authenticated key agreement protocol may also be adapted for
27 symmetric key implementations as shown in figure 6. In this case, as in figure 3
28 above, the key generation is omitted as the correspondents have a shared key obtained
29 over a secure channel.

30

